UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/085,839	02/25/2002	Michael A. Kozuch	42P10794	1747

45209 7590 02/25/2009
INTEL/BSTZ
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP
1279 OAKMEAD PARKWAY
SUNNYVALE, CA 94085-4040

EXAMINER	
GEE, JASON KAI YIN	

ART UNIT	PAPER NUMBER
2434	

MAIL DATE	DELIVERY MODE
02/25/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/085,839	**Applicant(s)** KOZUCH ET AL.	
	Examiner JASON K. GEE	**Art Unit** 2434	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *26 January 2009*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-7,9-16,18-27,29-31,39-44,46 and 47* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-7,9-16,18-27,29-31,39-44,46 and 47* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *25 February 2002* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. This action is response to communication: RCE filed on 01/26/2009.

2. Claims 1-7, 9-16, 18-27, 29-31, 39-44, and 46-47 are currently pending in this application. Claims 1, 12, 21, and 39 are independent claims.

3. No new IDS has been received for this application.

4. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 01/26/2009 has been entered.

Response to Arguments

5. Applicant's arguments filed 01/26/2009 have been fully considered but they are not persuasive.

In regards to the claim rejections, the applicants have argued that the Examiner has not established a prima facie case of obviousness. However, as seen in the rejections below, motivation to combine the references are provided in the art rejection found below.

Claim Rejections - 35 USC § 112

6. The previous 112 2nd paragraph rejections have been withdrawn in response to applicant's amendments.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 1-7, 9, 11-15, 17-25, 30, 31, 39-44, and 46-47 are rejected under 35 U.S.C. 103(a) as being obvious over England et al. US Patent No. 6,938,164 (hereinafter England), in view of Bruce Schneier's *Applied Cryptography* (Second Edition), and further in view of Fries US Patent No. 7,036,023 (hereinafter Fries).

As per independent claim 1, England teaches a method of loading a trustable operating system comprising:

Performing a start secure operation by a first processor of a plurality of processors (col. 4 lines 25-56); performing a join secure operation by remaining processors of the plurality of processors excluding the first processor, the join secure operation performed automatically (col. 4 lines 43-54) from the start secure operation and forces the remaining processors of the plurality of processors to enter into a halted state that prevents the remaining processors of the plurality of processors from interfering with the operations of the first processor (col. 4 lines 44-54; col. 10 lines 20-28; col. 10 lines 55-68; col. 13 lines 42-45); receiving signals by the first processor that the remaining processors have entered the halted stated (col. 4 lines 43-55); identifying a region in a memory of a computer by a one of a plurality of processors (col. 2 lines 16-21; col. 4 line 54 to col. 5 line 12; col. 3 lines 9-13); loading a content into the identified region (col. 5 lines 5-20); registering an identity of the content after the content is loaded onto the identified region, the registering comprises: recording a hash digest of the content of the identified region (col. 5 lines 49-65), the signed hash digest being stored in a register in the memory of the computer that is accessible by an outside entity to verify whether the content can be trusted (col. 5 lines 49-65; col. 14 lines 10-24; also col. 8 lines 14-50, wherein); and causing the one processor to jump to a known entry point in the content (col. 11 line 63 to col. 12 line 20); and completing the start secure operation by the first processor and signaling the remaining processors to resume activity by exiting the halted state and jumping to the known entry point in the identified region in the memory (col. 13 lines 42-61). England also teaches that a digest is signed in col. 14 lines 18-25. As a digest is signed, it is inherent that a hash signing engine is available. (England col. 14 lines 10-25 teaches that a signed certified digest exists.

More information can be found in Schneier pages 38 and 39, where a hash is signed (encrypting a hash with a private key is a method of signing).)

However, England does not explicitly teach that a hash digest is accessed via a secure channel. It does indeed teach that the microcode is included in a trusted core in col. 14 lines 10-25. The details of this trusted core is taught in England col. 5 lines 20-

48, and lines 42-48. These lines teach the extraction of this information, which occurs in a protected method. Extracting information via a protected method may be considered as accessing information via a secured channel.

Accessing important information via a secure channel is well known in the art, as can be seen by Fries in col. 5 lines 40-46 and col. 9 lines 45-59. Also, as seen in England, it would be obvious to verify a hash digest that is accessible by an outside entity. This is taught in col. 8 liens 14-50, where cryptographic measures may be stored on publicly accessible areas, in which they may be verified. These are directed toward hash digests, as seen in this section.

Although England does not explicitly teach wherein signals are received by the first processor *from the remaining processors* that the remaining processors have been halted, and loading a content *after* this signal is received, this would have been obvious. As seen in Figure 1 in England, the processors are all connected via a processor bus. They are able to communicate with one another. England, as taught throughout the reference, only allows the system to operate when one cpu is active, and the others are in a halted state. This is the core of the invention. Although England does not explicitly teach where a signal is received from informing the remaining processors are in a halted state, England's invention operates when only one cpu is active and the others are halted. It would have been obvious to do so, and there would be no core difference to the invention. Further, as England's system functions when one cpu is active and the others are halted, it would thus also infer that contents are loaded into the identified region after the operating cpu is informed that the other cpu's are halted. Loading these instructions are taught throughout England, such as in col. 5 lines 14-20.

At the time of the invention, it would have been obvious to one of ordinary skill in the art to include signing a hash digest in a system that registers an identity using a hash digest protocol. One of ordinary skill in the art would have been motivated to perform such an addition to save time. Schneier dictates on page 38 that "To save time, digital signature protocols are often implemented with on-way hash functions…. Instead of signing a document, Alice signs the hash of the document." Also, England indicates that additional information on hashing can be found in Schneier: "the reader is directed to a text written by Bruce Schneier and entited "Applied Cryptography: Protocols, Algorithms, and Source Code in C," published by John Wiley & Sons with copyright 1994 (or second edition with copyright 1996)" (col. 5 lines 61-65), and therefore, it is obvious to combine the teachings of Schneier's Applied Cryptography.

At the time of the invention, it would have been obvious to one of ordinary skill in the art to access secure information such as signatures via a secure channel. England already teaches storing secure information on a trusted core, such as in col. 5 lines 20-48, and extracting the information securely. One of ordinary skill in the art would be motivated to transport important information such as signed hashes via secured channels to provide security for the system, as sending signatures without a security channel compromises the security of the data as well as the system.

As per claim 2, preventing interference with the identifying, loading, and registering by at least one of a remaining one of the plurality of processors are taught in col. 2 lines 11-25. England dictates “According to one aspect, a memory controller prevents CPUs and other I/O bus masters from accessing memory during a code (for example, OS, microkernel, or other trusted core) initialization process.” Since this method prevents CPUs to access memory, it will restrict it from identifying, loading, and registering as memory cannot be accessed. Identifying, loading, and registering is taught in col. 9 line 57 to col. 10 line 11.

As per claim 3, preventing interference comprises halting at least the second processor of the plurality of processors until the identifying, loading, and registering is complete (col. 2 lines 10-25). The other processors are halted as they are being reset. England dictates “Once an initialization process has been executed by that CPU, the code is operational and any other CPUs are allowed to access memory (after being reset), as are any other bus masters (subject to any controls imposed by the initiated code).” Identifying, loading, and registering is taught in col. 9 line 57 to col. 10 line 11.

As per claim 4, causing at least the second processor of the plurality of processors to jump to the known entry point in the content is taught in col. 2 lines 10-25, as it states that “other CPUs are allowed to access memory” after the initialization process is complete.

As per claim 5, identifying comprises receiving a region parameter, the region parameter specifying a location of the region. This is taught in col. 9 lines 59-62, where it indicates that the start parameter “refers to the location in memory 110 where trusted core 146 begins (e.g., the memory address of the first instruction of platform trusted code portion 148).”

As per claim 6, the location comprises a range of addresses in the memory of the computer within which the region is located. Addresses are taught already in the rejection for claim 5 above, and can also be found in col. 11 line 63 to col. 12 line 20 and also col. 14 lines 55-65. Also, col. 10 line 55 to col. 11 line 15 indicate a range of addresses that can be accepted.

As per claim 7, the location comprises a start address and a length of the memory of the computer within which the region is located. This is taught in col. 9 lines 57-67: “the Trusted Core Initialization command includes three parameters: start, length of code, and length of memory … the start parameter refers to the location in memory where trusted core begins (e.g., the memory address of the first instruction…”.

As per claim 9, England teaches that the content is a component of an operating system to operate the computer. Col. 5 lines 13-20 indicate this: “Various components 144 of an operating system are thus laded into memory 110 … .”

As per claim 11, England teaches that loading and registering are uninterruptible. Col. 14 lines 45-54 indicate that interrupts are disallowed during the initialization process. It is already rejected in the above claims that the initialization process comprises loading and registering.

As per independent claim 12, England teaches an article of manufacture comprising: a machine-accessible medium including a data that, when accessed by a machine cause the machine to, halt all but one of a plurality of central processing units (CPU) in a computer (col. 2 lines 10-25; col. 3 lines 9-13); identify a region in a memory of the computer (col. 2 lines 16-21 and col. 4 lines 54 to col. 5 line 12); block access to the identified region by all resources except the non-halted CPU (col. 2 lines 10-25; col. 10 line 55 to col. 11 line 15); load a content into the identified region (col. 5 lines 5-20); registering an identity of the content of the identified region, the registering comprises: compute the cryptographic hash of the identified region (col. 5 lines 49-65; col. 14 lines 10-24; also Schneier); recording the computed cryptographic hash of the content in the identified region (col. 5 line 49 to col. 6 line 5; col. 14 lines 10-24, wherein it teaches that a cryptographically signed digest is retrieved, which would inherently indicate that a hash was recorded); and signing the computed cryptographic hash with a hash signing engine having a secure channel to access the cryptographic hash (rejected using the same arguments as claim 1), the signed cryptographic hash being stored in a register in the memory of the computer that is accessible to a third party to verify whether the content can be trusted (rejection seen in claim 1); and cause the non-halted CPU to being executing at a known entry point in the identified region after the identity of the content has been registered (col. 11 line 63 to col. 12 line 20). Also, the other limitations taught in this claim are rejected using the same basis of arguments used to reject claim 1 above.

As per claim 13, England teaches that the data that causes the machine to halt all but one of a plurality of CPUs comprises data causing the all but one of a plurality of CPUs to enter a halted state: "CPUs can be prevented form issuing read and write requests on processor bus 112, or by issuing a halt (e.g., HLT) command to the CPUs which halts the operation of each CPU until it resets" (col. 10 lines (62-67).

As per claim 14, England teaches that the data further causes the halted CPUs to exit the halted state after the non-halted CPU has begun executing at the known entry point in the identified region. This is taught in col. 2 lines 10-25, where the CPUs are reset and allowed to access memory through the initiated code. Also, this occurs after one of the plurality of CPUs has begun executing at the known entry point, as taught in col. 2 lines 10-25.

As per claim 15, England teaches in col. 2 lines 10-25 that the data further causes the previously hated CPUs to begin executing at the known entry point in the identified region upon exiting the halted state: "Once an initialization process has been executed by that CPU, the code is operational and any other CPUs are allowed to access memory (after being reset), as are any other bus masters."

Claim 18 is being rejected using the same basis of arguments used to reject claim 5.

Claim 19 is being rejected using the same basis of arguments used to reject claim 6.

Claim 20 is being rejected using the same basis of arguments used to reject claim 7.

As per independent claim 21, England teaches halting all but one of plurality of central processing units in a computer (col. 2 lines 10-25). Blocking access to the identified region by all resources except the non-halted CPU is taught in col. 10 line 55 to col. 11 line 15. Recording a cryptographic hash of the content of the identified region is taught in col. 5 line 49 to col. 6 line 5. Placing the non-halted CPU into a known privileged state is taught in col. 6 lines 38-63. Signing the cryptographic hash with a digest signing engine coupled to the memory of the computer having a secure channel to access the cryptographic hash and storing the hash in a register in the memory that is accessible by an outside identity to verify whether the content can be trusted is rejected using the same basis of arguments used to reject claim 1.

As per claim 22, jumping to a known entry point in the region is taught in (col. 11 line 63 to col. 12 line 20).

As per claim 23, England teaches that halting comprises causing the all but one of a plurality of CPUs to enter a special halted state (col. 10 line 55 to col. 11 line 15). This halted state is special as only the CPUs that need to be halted receive this ‘HLT’ command.

As per claim 24, England teaches exiting the special halted state after the non-halted CPU has begun executing at the known entry point in the identified region. This is taught in col. 2 lines 10-25, where the CPUs are reset and allowed to access memory through the initiated code.

As per claim 25, England teaches in col. 2 lines 10-25 that the data further causes the previously hated CPUs to begin executing at the known entry point in the identified region upon exiting the special halted state: “Once an initialization process has been executed by that CPU, the code is operational and any other CPUs are allowed to access memory (after being reset), as are any other bus masters.”

Claim 29 is being rejected using the same basis of arguments used to reject claim 5. This location is secured, as taught in col. 9 lines 62-67.

Claim 30 is being rejected using the same basis of arguments used to reject claim 6. Col. 11 line 63 to col. 12 line 20 indicate that the addresses are of the trusted core, which is secure.

Claim 31 is being rejected using the same basis of arguments used to reject claim 7. This region is secured, as this is part of the initialization sequence by the trusted core.

As per independent claim 39, England teaches a method of loading a trustable operating system comprising: selecting an area in a memory accessible to a processor (col. 2 lines 10-25); loading a data into the selected area (col. 5 lines 13-30); registering an identity of the data loaded in the selected area (as shown in rejection to claim 12); recording a unique cryptographic function of the data loaded in the selected area (as shown in the rejection for claim 12); signing the unique cryptographic function with a hash signing engine having a secure channel to access the unique cryptographic function, the signed unique cryptographic function being stored in a register in memory and accessible by an outside entity to verify whether the data is trustworthy (as shown in the rejection for claim 12 and 1); directing the processor to commence processing at an entry point in the selected area (col. 11 line 63 to col. 12 line 20); and preventing interruption of the selecting, load, registering, recording, signing, and directing until they are completed (col. 2 lines 11-24; since this method prevents CPUs to access memory until the first processor finishes initializing and the other CPUs are reset, it will prevent the selecting, loading, directing, registering, recording, and signing until they are completed. The selecting, loading, directing, registering, recording, and signing are part of the initialization process). Further, claim 39 is rejected using the same basis of arguments used to reject claim 1 above.

As per claim 40, halting any other processors having access to the memory until the selecting, loading, and directing is complete is taught in col. 10 line 55 to col. 11 line 15. These processors are halted until they are reset, and they are reset after the initialization process is complete, as indicated in col. 2 lines 10-25.

As per claim 41, causing the other processors to commence processing at an entry point in the selected area is taught in col. 2 lines 10-25, where the other CPUs are allowed to access the memory. It is also taught in col. 10 line 55 to col. 11 line 15 that the other CPUs are allowed to access the memory within an address range of the trusted core memory.

As per claim 42, receiving a parameter specifying a location of the area to be selected is taught in col. 9 lines 59-62, where it indicates that the start parameter "refers to the location in memory 110 where trusted core 146 begins (e.g., the memory address of the first instruction of platform trusted code portion 148)."

As per claim 43, the location comprises a range of addresses in the memory of the computer within which the region is located. Addresses are taught already in the rejection for claim 5 above, and can also be found in col. 11 line 63 to col. 12 line 20 and also col. 14 lines 55-65. Also, col. 10 line 55 to col. 11 line 15 indicate a range of addresses that can be accepted.

As per claim 44, the location comprises a start address and a length of the memory of the computer within which the region is located. This is taught in col. 9 lines 57-67: "the Trusted Core Initialization command includes three parameters: start, length of code, and length of memory … the start parameter refers to the location in memory where trusted core begins (e.g., the memory address of the first instruction…".

Claim 46 is rejected using the same basis of arguments used to reject claim 9. The memory resides in this device, as it is an internal memory, as indicated in col. 5 lines 5-12.

As per claim 47, England indicates in col. 5 lines 13-20 that the operating systems can be Windows® operating systems. It is inherent that Windows® has a graphical user interface.

9. Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over the England combination as applied above, and further in view of ATPM – Review: Virtual PC 4.0 (April 2001), by Gregory Tetrault.

As per claim 10, col. 5 lines 13-20 indicate various operating systems. A privileged software nucleus is taught in col. 6 lines 38-63, in which different privilege levels are taught. However a virtual machine monitor is not taught in England. However, this is taught in ATPM's review of Virtual PC, reviewed by Gregory Tetrault. ATPM indicates that Windows® supported virtual machine.

At the time of the invention, it would have been obvious to one of ordinary skill in the art to include the use of virtual machines when using Windows® operating systems. One would have been motivated to perform such an addition, because a virtual machine is an option provided by operating systems such as Windows®, and is useful for networking. Col. 3 lines 4-14 of England indicates that the invention can apply to network PCs as well, and a virtual machine is an example of a network PC.

10. Claim 16, 26, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over the England combination as applied above, and further in view of England et al. US Patent No. 6,330,670 (hereinafter '670).

As per claim 16, a required platform information is recorded in the hash digest area, as England indicates that the digest contains a value that can be considered to uniquely represent the trusted core in use. Erasing a hash digest area (which is a register) is taught in col. 12 lines 31-39. This section teaches that a CPU clears the states of the CPU, which is located in the registers. However, at the time of the invention, ‘670 does not explicitly teach wherein the platform information includes a version number of the one of the plurality of CPU’s. This is taught by ‘670 though, such as in col. 2 line 60 to col. 3 line 15.

At the time of the invention, it would have been obvious to one of ordinary skill in the art to implement checking platform information for authorization. One of ordinary skill in the art would have been motivated to perform such an addition to create a secure environment for operating systems to be executed on, focusing on the correct and authorized hardware components trusted to run such information. As shown in the passage, this is well known in the art, and is used to ensure that software is running on the correct hardware, thereby increasing security.

Claim 26 is rejected using the same basis of arguments used to reject claim 16.

Claim 27 is rejected using the same basis of arguments used to reject claim 17.

Conclusion

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to JASON K. GEE whose telephone number is (571)272-6431. The examiner can normally be reached on M-F, 7:00 am to 4:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner’s supervisor, Kambiz Zand can be reached on (571) 272-3811. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Jason Gee
Patent Examiner
Technology Center 2400
02/17/2009
/Kambiz Zand/

Supervisory Patent Examiner, Art Unit 2434